BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

WHAT IS CLAIMED IS:

1. A microprocessor comprising:

a predecode unit configured to receive instruction bytes and generate corresponding predecode information;

an instruction cache coupled to the predecode unit and configured to store the instruction bytes and the predecode information corresponding to the instruction bytes,

a load/store unit configured to receive data bytes;

a data cache configured to receive and store the data bytes from the load/store unit, and

a level two cache configured to receive and store victimized instruction bytes from the instruction cache and victimized data bytes from the data cache, wherein the level two cache is configured to receive and store parity information and predecode information for the stored victimized instruction bytes and wherein the level two cache is configured to receive and store error correction code bits for the stored victimized data bytes.

2. The microprocessor of claim 1, further including parity generation and checking logic configured to:

generate the parity bits for the instruction bytes transferred to the level two cache, and check the parity bits for the instruction bytes transferred from the level two cache.

4 3. The microprocessor of claim 1, further including error checking and correction logic configured to:

generate the error correction code bits for the data bytes transferred to the level two cache, and

check the error correction code bits for the data bytes transferred from the level two cache.

5 4 4. The microprocessor of claim 3, wherein the error checking and correction logic is configured to use the error correction code bits to correct at least one bit errors in the data bytes transferred from the level two cache.

3 8. The microprocessor of claim 1, wherein the level two cache is divided into cache lines, wherein the cache lines include a first storage area and a second storage area, wherein each cache line is configured to store one or more indicator bits indicative of whether: (a) instruction bytes are stored in the cache line's first storage area, and predecode bits and a parity bit are stored in the cache line's second area, or (b) data bytes are stored in the cache line's first storage area and error correction code bits are stored in the cache line's second storage area.

6. An apparatus comprising:

a processor configured to receive instruction bytes and data bytes, wherein the processor is configured to operate on the data bytes according to instructions formed by the instruction bytes; and

a cache configured to receive and store victimized instruction bytes and victimized data bytes from the processor, wherein the cache is configured to receive and store parity information and predecode information for the stored victimized instruction

bytes, and wherein the cache is configured to receive and store error correction code (ECC) bits for the stored victimized data bytes.

7. The apparatus of claim 6, wherein the cache is configured to provide the victimized data bytes and the corresponding stored ECC bits to the processor in response to the processor requesting the victimized data bytes.

11
8. The apparatus of claim 6, wherein the cache is configured to provide the victimized instruction bytes and the corresponding stored parity information and the corresponding stored predecode information to the processor in response to the processor requesting the victimized instruction bytes.

12 11
9. The apparatus of claim 8, wherein the processor is configured to use the conveyed predecode information in lieu of generating new predecode information.

8
10. The apparatus of claim 6, wherein the cache is a level-two cache that is implemented on a common die with the processor.

9
11. The apparatus of claim 6, wherein the cache is a level-two cache that is implemented on a different die than the processor.

13
12. The apparatus of claim 6, wherein the cache comprises cache lines, wherein each cache line is configured to store an indicator bit indicative of whether the logical block stores predecode bits for instruction bytes or error checking and correction bits for data bytes.

14 13
~~13~~. The apparatus of claim ~~12~~, wherein each cache line include a first storage area and a second storage area, wherein each cache line is configured to store an indicator bit indicative of whether: (a) instruction bytes are stored in the cache line's first storage area, and predecode bits and a parity bit are stored in the cache line's second area, or (b) data bytes are stored in the cache line's first storage area and ECC bits are stored in the cache line's second storage area.

10
~~14~~. The apparatus of claim 6, wherein the processor further comprises parity generation and checking logic configured to:

generate the parity bits for the instruction bytes transferred to the cache, and

check the parity bits for the instruction bytes transferred from the cache.

15. The apparatus of claim 6, wherein the processor further comprises error checking and correction logic configured to:

generate the ECC bits for the data bytes transferred to the cache, and

check the ECC bits for the data bytes transferred from the cache.

16. The apparatus of claim 15, wherein the error checking and correction logic is configured to use the ECC bits to correct at least one bit errors in the data bytes transferred from the cache.

17. A method comprising:

receiving instruction bytes;

generating predecode information for the instruction bytes;

storing the instruction bytes and the predecode information in a first memory;

outputting at least a portion of the instruction bytes and the predecode information with parity information to a second memory in response to the instruction bytes being overwritten in the first memory;

receiving data bytes;

storing the data bytes to a third memory; and

outputting at least a portion of the data bytes with corresponding error correction code information to the second memory in response to the data bytes being overwritten in the third memory.

18. The method of claim 17, wherein the second memory is divided into cache lines, wherein the method further comprises storing an indicator bit in each cache line in the second memory, wherein the indicator bit is indicative of whether the predecode bits or the error correction code bits are stored therein.

19. The method of claim 18, further comprising:
storing an indicator bit for each logical block in the second memory, wherein the indicator bit is indicative of whether predecode bits or error checking and correction bits are stored therein.

20. The method of claim 18, further comprising:
conveying the at least one stored parity bit and the corresponding stored portion of the instruction bytes and the predecode information back to the processor, wherein the

processor is configured to use the conveyed predecode information in lieu of generating new predecode information if the parity bit is correct.

21. A computer system comprising:

a main system memory;

a memory controller coupled to the main system memory;

a microprocessor coupled to the memory controller, wherein the microprocessor is configured to receive instruction bytes and data bytes, wherein the processor is configured to operate on the data bytes according to instructions formed by the instruction bytes; and

a cache configured to receive and store victimized instruction bytes and victimized data bytes from the processor, wherein the cache is configured to receive and store parity information and predecode information for the stored victimized instruction bytes, and wherein the level two cache is configured to receive and store error correction code bits for the stored victimized data bytes.

22. The computer system of claim 21, wherein the memory controller further comprises parity generation and checking logic configured to generate and check parity for bytes transferred to and from the main system memory.

23. The computer system of claim 21, wherein the memory controller further comprises error correction code generation and checking logic configured to generate and check error correction code bits for bytes transferred to and from the main system memory.